NMAM INSTITUTE OF TECHNOLOGY

(An Autonomous Institution affiliated to VTU, Belgaum)

(AICTE approved, NBA Accredited, ISO 9001:2008 Certified)

Nitte – 574110, Karkala, Udupi District, Karnataka, India.

Department of Computer Science and Engineering

13CS603 - Java and Internet Technologies

Marks Distribution

Execution (Part A + Part B)	15+15
Write up	5+5
Viva	10
Change of one question	-5

PART-A

1. **Write a java program for generating 2 threads, one for printing even numbers and the other for printing odd numbers.**

```
import java.io.*;

classeventhread implements Runnable{
        public void run(){
                for(int j=2;j<=100;j+=2)
                        System.out.println("even="+j);
        }
}
class oddthread implements Runnable{
        public void run(){
                for(int j=1;j<=100;j+=2)
                        System.out.println("odd="+j);
        }
}

public class ques1{
        public static void main(String args[]){
                oddthread t1=new oddthread();
                eventhread t2=new eventhread();
                Thread t1obj=new Thread(t1);
                Thread t2obj=new Thread(t2);
                t1obj.start();
                t2obj.start();
        }
}
```

2. **Create a base class and derived class having data and overridden function members. Demonstrate the dynamic method dispatch.**

```
class Figure{
        double dim1;
        double dim2;
        Figure(double a, double b){
                dim1 = a;
                dim2 = b;
        }
        double area(){
                System.out.println("Area for Figure is undefined.");
                return 0;
        }
}

class Rectangle extends Figure{
        Rectangle(double a, double b){
                super(a, b);
        }
        // override area for rectangle
        double area(){
                System.out.println("Area of Rectangle.");
                return dim1 * dim2;
        }
}

class Triangle extends Figure{
        Triangle(double a, double b){
                super(a, b);
        }
        // override area for right triangle
        double area(){
                System.out.println("Area of Triangle.");
                return dim1 * dim2 / 2;
        }
}

class dynamicdisp{
        public static void main(String args[]){
                Figure f = new Figure(10, 10);
                Rectangle r = new Rectangle(9, 5);
                Triangle t = new Triangle(10, 8);
                Figure figref;
                figref = r;
                figref.area();
                figref = t;
                figref.area();
                figref = f;
                figref.area();
        }
}
```

3. **Create a class customer for a bank. Class account should inherit the properties of customer and having 'type of account' and 'interest rate' as members. Use constructors in base and derived class to initialize the data members. An account may be created with zero initial deposit. (Use default arguments, super keyword). Implement the methods Deposit (), Withdraw (), and Get_Balance() to operate on account**.

```
import java.io.*;
import java.util.*;

class customer{
        String name;
        Long accno;
        double balance;
        customer(String s,long a){
                name=s;
        accno=a;
        balance=0;
    }
    void display(){
        System.out.println("\nName="+name+"\nAccount_no="+accno+"\nBalance="+balance);
    }
}

class account extends customer{
    String acc_type;
    Double intrst_rt;
    account(String s,longaccno,String at){
        super(s,accno);
        acc_type=at;
    }
    void Deposit(double amt){
        balance+=amt;
    }
    void Withdraw(double amt){
        if((balance-amt)<=0)
            System.out.println("Amount for withdraw is less than balance");
        else
            balance=balance-amt;
    }
    doubleGetBalance(){
        return(balance);
    }
    void display(){
        super.display();
        System.out.println("Account_type="+acc_type);
    }
}
class Bank{
    public static void main(String[] args){
        boolean flag=true;
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the name of customer");
        String name=s.nextLine();
        System.out.println("Enter the account number");
        longacc=s.nextInt();
        System.out.println("Enter the account type");
        s.nextLine();
        String acct=s.nextLine();
        account a1=new account(name,acc,acct);
        while(flag){
            System.out.println("Press 1:Deposit\n2:withdraw\n3:GetBalance\n4:display\n5:exit");
            System.out.println("Enter option");
            int c=s.nextInt();
            switch(c){
                case 1:System.out.println("Enter the amount");
                        double amount=s.nextDouble();
                        a1.Deposit(amount);
                        break;
```

```
                                        case 2:System.out.println("Enter the amount to be withdrawn");
                                                        double at=s.nextInt();
                                                        a1.Withdraw(at);
                                                        break;
                                        case 3:double b=a1.GetBalance();
                                                        System.out.println("Balance="+b);
                                                        break;
                                        case 4:a1.display();
                                                        break;
                                        case 5:flag=false;
                                }
                        }
                }
}
```

4. **Create a class as Student. Write a program in Java to display the names and roll numbers of students. Create an array of 10 students and initialize the array with user input. Handle ArrayIndexOutOfBoundsExeption, so that any such problem doesn't cause illegal termination of program. Read a character from user and display the student names starting with given character.**

```
import java.io.*;
import java.util.*;

class func {
        String name;
        int roll;
        public void read() {
                Scanner s1=new Scanner(System.in);
                System.out.println("enter Student name is ");
                name=s1.nextLine();
                System.out.println("enter Student roll is ");
                roll=s1.nextInt();
        }
        public void display() {
                System.out.println("Student name is "+ name);
                System.out.println("Student roll is "+ roll);
        }
}
public class stuk {
        public static void main(String args[])throws Exception {
                int n;
                System.out.println("enter the no of students");
                Scanner s=new Scanner(System.in);
                n=s.nextInt();
                func ob[]=new func[10];
                try {
                        for(int i=0;i<n;i++) {
                                ob[i]=new func();
                                ob[i].read();
                        }
                }
                catch(ArrayIndexOutOfBoundsException e) {
```

```
            System.out.println(e);
        }
        System.out.println("enter a char");
        BufferedReader b=new BufferedReader(new InputStreamReader(System.in));
        //to read a character
        char t=(char)b.read();
        int f=0;
        for(int i=0;i<n;i++) {
            String sec=ob[i].name;
            if(sec.charAt(0)==t) {
                f=1;
                ob[i].display();
                break;
            }
        }
        if(f==0) {
            System.out.println("not found");
        }
    }
}
```

5. **Create a class as Student. Write a program in Java to display the names and roll numbers of students in sorted order. Place the read and display functions in a package. Handle any possible exception, so that any such problem doesn't cause illegal termination of program.**

```
import p2.*;
importjava.util.*;

public class stu{
    public static void main(String args[]){
        int n;
        System.out.println("enter the no of students");
        Scanner s=new Scanner(System.in);
        n=s.nextInt();
        studentob[]=new student[2];
        try{
            for(inti=0;i<n;i++){
                ob[i]=new student();
                ob[i].read();
            }
        }
        catch(ArrayIndexOutOfBoundsException e){
            System.out.println(e);
        }
        student temp;
        for(inti=0;i<n;i++){
            for(int j=0;j<n-i-1;j++){
                if(ob[j].roll>ob[j+1].roll){
                    temp=ob[j];
                    ob[j]=ob[j+1];
```

```
                                        ob[j+1]=temp;
                                }
                        }
                }
                System.out.println("sorted list is");
                for(inti=0;i<n;i++){
                        ob[i].display();
                }
        }
}
```

```
/*p2 package*/
package p2;
importjava.util.*;

public class student{
        public String name;
        public int roll;
        public void read(){
                Scanner sc=new Scanner(System.in);
                System.out.println("enter a name");
                name=sc.nextLine();
                System.out.println("enter the roll");
                roll=sc.nextInt();
        }
        public void display(){
                System.out.println("name:"+name+" rollno:"+roll);
        }
}
```

6. **Define a Person class with three data members’ age, name and gender. Derive a class called Employee from Person that adds a data member 'emp_code' to store employee code. Derive another class called Specialist from Employee. Add a method to each derived class to display the information about what it is. Write a program to generate an array of three ordinary employees and another array of three specialists and display the information about them. Also display the information of the specialists by calling the method inherited from employee class.**

```
import java.io.*;

class person {
        String name;
        int age;
        String gender;
}

class employee extends person{
        int code;
        void set_e()throws IOException{
                DataInputStream d =new DataInputStream(System.in);
                String s1,s2;
                System.out.println("\nENTER NAME ");
                name=d.readLine();
                System.out.println("ENTER AGE ");
                s1=d.readLine();
                age=Integer.parseInt(s1);
                System.out.println("ENTER GENDER ");
                gender=d.readLine();
                System.out.println("ENTER CODE ");
                s2=d.readLine();
                code=Integer.parseInt(s2);
        }
        void disp_e()throws IOException{
                System.out.println("NAME :"+this.name);
                System.out.println("AGE :"+this.age);
                System.out.println("GENDER :"+this.gender);
                System.out.println("CODE "+this.code);
        }
}

class spec extends employee{
        void set_s()throws IOException{
                DataInputStream d =new DataInputStream(System.in);
                String s1,s2;
                System.out.println("\nENTER NAME ");
                name=d.readLine();
                System.out.println("ENTER AGE ");
                s1=d.readLine();
                age=Integer.parseInt(s1);
                System.out.println("ENTER GENDER ");
                gender=d.readLine();
                System.out.println("ENTER CODE ");
                s2=d.readLine();
                code=Integer.parseInt(s2);
        }
        void disp_s()throws IOException{
                System.out.println("NAME :"+this.name);
                System.out.println("AGE :"+this.age);
                System.out.println("GENDER :"+this.gender);
                System.out.println("CODE "+this.code);
        }
}

class PersonDemo{
        public static void main(String args[])throws IOException{
                int i;
                employee e[] = new employee[3];
                spec s[] = new spec[3];
                for(i=0;i<3;i++){
                        e[i]= new employee();
                        System.out.println("ENTER EMPLOYEE DETAILS\n");
                e[i].set_e();
            }
            for(i=0;i<3;i++){
                s[i]= new spec();
                System.out.println("ENTER SPECIALIST DETAILS\n");
                s[i].set_s();
            }
            for(i=0;i<3;i++){
                System.out.println("");
                System.out.println("\n EMPLOYEE "+(i+1)+" DETAILSA\n");
                e[i].disp_e();
            }
            for(i=0;i<3;i++){
                System.out.println("");
                System.out.println("SPECIALIST "+(i+1)+" DETAILS\n");
                s[i].disp_s();
            }
        }
}
```

7. **Write a frame program that inputs the three floating point numbers from the user and display the sum, avg, largest of three these numbers.**

```
import java.awt.*;
import javax.swing.*;

class calc implements ActionListener{
        JLabel l1=new JLabel("First Number");
        JLabel l2=new JLabel("Second Number");
        JLabel l3=new JLabel("Third Number");
        JTextField t1=new JTextField(10);
        JTextField t2=new JTextField(10);
        JTextField t3=new JTextField(10);
        JButton b1=new JButton("Sum");
        calc(){
                JFrame f=new JFrame("calc");
                f.setLayout(new FlowLayout());
                f.setSize(500,500);
                f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

                f.add(l1);
                f.add(t1);
                f.add(l2);
                f.add(t2);
                f.add(l3);
                f.add(t3);
                f.add(l4);
                f.add(t4);
                f.add(b1);
                b1.addActionListener(this);
                f.setVisible(true);
        }
        public void actionPerformed(ActionEvent ae) {
                float a=Float.parseFloat(t1.getText());
                float b=Float.parseFloat(t2.getText());
                float c=Float.parseFloat(t3.getText());
                if(ae.getActionCommand().equals("Sum"));{
                        float s=a+b+c;
                        float t=((a+b+c)/3);
                        float w=(Math.max(Math.max(a, b),c));
                        JOptionPane.showMessageDialog(null, "sum is"+s+"\navg is"+t+"\nmax is"+w);
                }
        }
}

public class calculater {
        public static void main(String args[]){
                calc h=new calc();
        }
}
```

8. **Consider the string "NMAMIT" [Nitte] (Autonomous Institute) {Karkala}. Write synchronized thread program to print this string. Demonstrate the importance of synchronization.**

```
class MyThread extends Thread{
        private String temp,leftmarker,rightmarker;
        public MyThread(String t,String l,String r){
                temp = t;
                leftmarker = l;
                rightmarker = r;
        }
        public void run(){
                display(temp,leftmarker,rightmarker);
        }
        synchronized public static void display(String temp,String leftmarker,String rightmarker){
                System.out.print(leftmarker);
                try{
                        Thread.sleep(100);
                }
                catch(Exception e){
                        System.out.println("Exception Ocuured "+e);
                }
                System.out.print(temp);
                System.out.println(rightmarker);
        }
}
```

```
class prgm8{
        public static void main(String args[]){
                MyThread t1 = new MyThread("NMAMIT","\"","\"");
                MyThread t2 = new MyThread("Nitte","[","]");
                MyThread t3 = new MyThread("Autonomous Institue","(",")");
                MyThread t4 = new MyThread("Karkala","{","}");
                try{
                        t1.start();
                        t2.start();
                        t3.start();
                        t4.start();
                }
                catch(Exception e){
                        System.out.println("Exception Ocuured ");
                }
        }
}
```

9. **Develop a set of methods, which work with an integer array. The methods to be implemented are:-**
   (i) min (which finds the minimum element in the array)
   (ii) max (which finds the maximum element in the array – variable length argument)
   (iii) scale (method to multiply the array – Use default arguments)
   Place this in a package called p1. Let this package be present in a folder called "myPackages", which is a folder in your present working directory. Write a main method to use the methods of package p1.

```
package arraymet_min;

import java.util.Scanner;

public class array {
        int a[],size;
        public array(int size){
                a=new int[size];
        }
        public void getarray(int n){
                int i;
                for(i=0;i<n;i++){
                        Scanner s = new Scanner(System.in);
                        a[i]=s.nextInt();
                }
        }
        public int min(int n){
                int small,i;
                small=a[0];
                for(i=1;i<n;i++){
                        if(a[i]<small){
                                small=a[i];
                        }
        }
        return(small);
    }
    public int max(int n){
        int max,i;
        max=a[0];
        for(i=1;i<n;i++){
            if(a[i]>max){
                max=a[i];
            }
        }
        return(max);
    }
    public void scale(int n,int x){
        int i;
        for(i=0;i<n;i++)
            a[i]=a[i]*x;
        System.out.println("Multiplied array");
        for(i=0;i<n;i++){
            System.out.println(a[i]);
        }
    }
}
```

```
import java.util.Scanner;
import arraymet_min.array;

public class pkk1 {
    public static void main(String args[]){
        int n,i,x;
        array a1=new array(50);
        System.out.println("Enter the size of the array");
        Scanner s=new Scanner(System.in);
        n=s.nextInt();
        System.out.println("Enter the array");
        a1.getarray(n);
        System.out.println("Minimum element in the array is:");
        int small=a1.min(n);
        System.out.println(small);
        System.out.println("Maximum element in the array is:");
        int large=a1.max(n);
        System.out.println(large);
        System.out.println("Enter the scalar to multiply:");
        Scanner s3=new Scanner(System.in);
        x=s3.nextInt();
        a1.scale(n,x);
    }
}
```

10. **Write a program to handle following exceptions: 1. Arithmetic 2.Array index out of Bound 3. Null pointer 4. Number Format 5. Illegal Access. Include one nested exception case.**

```
import java.util.Scanner;

class ExceptionTest{
	void arithmeticException(){
		try{
			int a=10;
			int b=5;
			int c=5;
			int x=a/(b-c);
			System.out.println("X= "+x);
			int y=a/(b+c);
			System.out.println("Y= "+y);
			//throw new ArithmeticException("ExceptionTest");
		}
		catch(ArithmeticException e){
			System.out.println(e);
		}
	}
	void ArrayIndexOutOfBoundException(){
		try{
			int a[]=new int[4];
			System.out.println(a[5]);
		}
		catch(ArrayIndexOutOfBoundsException e){
			System.out.println(e);
		}
	}
	void nullPointerexception(){
		try{
			System.out.println("This is invalid use of null reference");
			throw new NullPointerException("test");
		}
		catch(NullPointerException e){
			System.out.println(e);
		}
	}
	void numberFormatException(){
		try{
			throw new NumberFormatException("NumberFormat");
		}
		catch(NumberFormatException e){
			System.out.println(e);
		}
	}
	void illegalAccessException(){
        try{
            throw new IllegalAccessException("Illegel");
        }
        catch(IllegalAccessException e){
            System.out.println(e);
        }
    }
}

public class ExceptionDemo{
    public static void main(String args[]){
        int choice;
        boolean c=true;
        Scanner sc=new Scanner(System.in);
        ExceptionTest Exc=new ExceptionTest();
        while(c){
            System.out.println("1:Arithmeticexception");
            System.out.println("2:ArrayIndexOutofBoundexception");
            System.out.println("3:Null pointer Exception");
            System.out.println("4:Numberformat Exception");
            System.out.println("5:Illegal Access");
            System.out.println("6:Exit");
            System.out.println("Enter your choice: ");
            choice=sc.nextInt();
            switch(choice){
                case 1:Exc.arithmeticException();
                        break;
                case 2:Exc.ArrayIndexOutOfBoundException();
                        break;
                case 3:Exc.nullPointerexception();
                        break;
                case 4:Exc.numberFormatException();
                        break;
                case 5:Exc.illegalAccessException();
                        break;
                case 6:c=false;
                        break;
                default:System.out.println("Invalid choice");
                        break;
            }
        }
    }
}
```

PART-B

11. **Create a frame/Swing that is having a List View to select the city, Text box to enter the name, phone number. Radio button to choose the gender and check box to choose hobbies. Create an alert to display the selections made by the user.**

```
import java.awt.*;
import javax.swing.*;

class A implements ActionListener{
        JFrame jfrm=new JFrame("GUI");
        String citi[]={"karwar","goa"};
        JList <String> li;
        JTextField t1;
        JRadioButton r1,r2;
        JCheckBox ch1,ch2;
        JLabel l1,l2,l3;
        ButtonGroup bg;
        JButton b;
        A(){
                jfrm.setLayout(new FlowLayout());
                jfrm.setSize(300,300);
                jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                l1=new JLabel("Enter city");
                li=new JList <String> (citi);
                l2=new JLabel("Enter name : ");
                t1=new JTextField(10);
                l3=new JLabel("Enter gender");
                r1=new JRadioButton("male");
                r2=new JRadioButton("female");
                bg=new ButtonGroup();
                bg.add(r1);
                bg.add(r2);
                ch1=new JCheckBox("prog");
                ch2=new JCheckBox("simulation");
                b=new JButton("save");
                b.addActionListener(this);
                jfrm.add(l1);
                jfrm.add(li);
                jfrm.add(l2);
                jfrm.add(t1);
                jfrm.add(l3);
                jfrm.add(r1);
                jfrm.add(r2);
                jfrm.add(ch1);
                jfrm.add(ch2);
                jfrm.add(b);
                jfrm.setVisible(true);
        }
        public void actionPerformed(ActionEvent e){
                int i;
                String s_name=null;
                String gender=null;
                String hob = null;
                String display=null;
                i=li.getSelectedIndex();
                s_name=t1.getText();
                if(r1.isSelected()){
                        gender="male";
                }
                if(r2.isSelected()){
                        gender="female";
                }
                if(ch1.isSelected()){
                        hob  = "prog";
                        if(ch2.isSelected()){
                                hob = hob+"simulation";
                        }
                }
                else{
                        if(ch2.isSelected()){
                                hob = "simulation";
                        }
                }
                display="city "+citi[i]+" name is "+s_name+" gender "+gender+" hobbies "+hob;
                JOptionPane.showMessageDialog(null,display);
        }
}

public class M {
  public static void main(String[] args){
          new A();
  }
}
```

12. Write a program to copy the contents of two files in to one file.

```
import java.util.*;
import java.io.*;

public class r1{
        public static void main(String args[]) throws IOException{
                PrintWriter p1=new PrintWriter(new File("w.txt"));
                Scanner f1 =new Scanner(new File("a.txt"));
                Scanner f2 =new Scanner(new File("b.txt"));
                while(f1.hasNext()){
```

```
            String s1 =f1.nextLine();
            p1.println(s1);
        }
        while(f2.hasNext()){
            String s1 =f2.nextLine();
            p1.println(s1);
        }
        f1.close();
        f2.close();
        p1.close();
    }
}
```

13. **Write a frame/swing program to display the contents of a file in a text area, once the user edits the contents, provide a button to save the contents back to same file.**

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.util.*;
import javax.swing.*;

class a implements ActionListener{
    JTextArea a1;
    JButton b1;
    Scanner s1;
    PrintWriter p1;
    //File f1;
    FileWriter f2;
    a(){
        try{
            s1=new Scanner(new File("w.txt"));
            txtar();
        }
        catch(Exception e){
        }
    }
    public void txtar(){
        JFrame j =new JFrame("tt");
        j.setLayout(new FlowLayout());
        j.setSize(300,300);
        a1=new JTextArea(10,10);
        b1= new JButton("save");
        b1.addActionListener(this);
        j.add(a1);
        j.add(b1);
        j.setVisible(true);
        stext();
    }
```

```
        public void actionPerformed(ActionEvent e){
                fwrite();
        }
        public void stext(){
                while(s1.hasNext()){
                        String ss =s1.nextLine();
                        a1.append(ss);
                }
        }
        public void fwrite(){
                try{
                        p1=new PrintWriter(new File("w.txt"));
                        p1.write(a1.getText());
                        p1.flush();
                        p1.close();
                }
                catch(Exception e){
                }
        }
}

public class textarea {
        public static void main(String args[]){
                a aa1 = new a();
        }
}
```

14. **Write a frame/Swing program that is having an radio buttons to draw circle, line, rectangle. Another set of radio buttons to choose the background color. The color of frame background should be set to the selected color.**

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class d extends JFrame implements ActionListener{
        JRadioButton d1,d2,d3,c1,c2,c3;
        JButton b1;
        int d;
        JFrame j;
        d(){
                //j= new JFrame("rr");
                setLayout(new FlowLayout());
                setSize(300,300);
                //setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                d1= new JRadioButton("circle");
                d2= new JRadioButton("rec");
                d3= new JRadioButton("line");
                d1.addActionListener(this);
                d2.addActionListener(this);
                d3.addActionListener(this);
                ButtonGroup b1 =new ButtonGroup();
                ButtonGroup b2 =new ButtonGroup();
                b1.add(d1);
                b1.add(d2);
                b1.add(d3);
                c1= new JRadioButton("red");
                c2= new JRadioButton("green");
                c3= new JRadioButton("blue");
                c1.addActionListener(this);
                c2.addActionListener(this);
                c3.addActionListener(this);
                b2.add(c1);
                b2.add(c2);
                b2.add(c3);
                add(d1);
                add(d2);
                add(d3);
                add(c1);
                add(c2);
                add(c3);
                setVisible(true);
                repaint();
        }
        public void actionPerformed(ActionEvent e){
                String cmd=e.getActionCommand();
                if(cmd=="circle"){
                    d=1;
                }
                else if(cmd=="rec"){
                    d=2;
                }
                else if(cmd=="line"){
                    d=3;
                    repaint();
                }
                if(cmd=="red"){
                    getContentPane().setBackground(Color.RED);
                }
                else if(cmd=="green"){
                    getContentPane().setBackground(Color.green);
                }
                else if(cmd=="blue"){
                    getContentPane().setBackground(Color.blue);
                }
        }
        public void paint(Graphics g){
                super.paint(g);
                if(d==1)
                    g.drawOval(100,100, 50, 50);
                if(d==2)
                    g.drawRect(100, 100, 50, 50);
                if(d==3)
                    g.drawLine(100, 100,300, 300);
        }
    }

    public class t {
        public static void main(String args[]){
                d dr=new d();
        }
    }
```

15. Write a JDBC program to create a table, insert data and then display all the records.

```
import java.io.*;
import java.util.*;
import java.sql.*;

class dbdemo {
    static Connection con;
    static Statement stm;
    static ResultSet rs;
    dbdemo() throws SQLException{
        try{
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://172.16.2.3/student","student","student");
        }
        catch(ClassNotFoundException e){
            System.out.println("Unable to load the driver.");
        }
        catch(SQLException e){
            System.out.println("Connection not established.");
        }
        stm = con.createStatement();
    }
    void displayinfo() throws SQLException {
        String s = "SELECT * FROM sanacc";

        rs = stm.executeQuery(s);

        while(rs.next()){


    System.out.println(rs.getInt(1)+"\t"+rs.getString(2)+"\t"+rs.getString(3)+"\t"+rs.getFloat(4));

        }

    }

    void insertinfo() throws SQLException{

        Scanner s = new Scanner(System.in);

        System.out.println("Enter the account number");

        int accno = s.nextInt();

        System.out.println("Enter surname");

        String sn = s.next();

        System.out.println("Enter firstname");

        String fn = s.next();

        System.out.println("Enter balance");

        float bn = s.nextFloat();

        String sq = "INSERT INTO sanacc VALUES("+accno+",'"+sn+"','"+fn+"',"+bn+")";

        int r = stm.executeUpdate(sq);

        System.out.println(r+" Row updated successfully ");

    }

    void createTable()throws SQLException{

        String sq = "create table sanacc( accno integer, surname varchar(15), firstname varchar(15), balance integer )";

        int r = stm.executeUpdate(sq);

        System.out.println("Table created successfully.");
    }
    void closecon()throws SQLException{
        rs.close();
        stm.close();
        con.close();
    }
}

public class sqlmethods {
    public static void main(String[] args) throws SQLException {
        boolean f = true;
        dbdemo d = new dbdemo();
        Scanner s = new Scanner(System.in);
        int ch;
        do{
            System.out.println("1.Create Table\n2.Display\n3.Insert\n4.Exit\nEnter your Choice");
            ch = s.nextInt();
            switch(ch){
                case 1: d.createTable();
                        break;
                case 2: d.displayinfo();
                        break;
                case 3: d.insertinfo();
                        break;
                case 4: f = false;
                        break;
                default: System.out.println("Invalid choice.");
            }
        }while(f);
        d.closecon();
    }
}
```

16. Write a JDBC program to create a table student having name and USN. Insert the data to the table and search the name of a given USN.

```
import java.io.*;
import java.util.*;
import java.sql.*;

class dbdemo {
    static Connection con;
    static Statement stm;
    static ResultSet rs;
    dbdemo() throws SQLException{
        try{
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://172.16.2.3/student","student","student");
        }
        catch(ClassNotFoundException e){
            System.out.println("Unable to load the driver.");
        }
        catch(SQLException e){
            System.out.println("Connection not established.");
        }
        stm = con.createStatement();
    }
    void displayinfo() throws SQLException {
        String s = "SELECT * FROM student";
        rs = stm.executeQuery(s);
        while(rs.next()){
            System.out.println(rs.getInt(1)+"\t"+rs.getString(2));
        }
    }
    void insertinfo() throws SQLException{
        Scanner s = new Scanner(System.in);
        System.out.println("Enter the USN");
        int usn = s.nextInt();
        System.out.println("Enter name");
        String name = s.next();
        String sq = "INSERT INTO student VALUES("+usn+",'"+name+")";
        int r = stm.executeUpdate(sq);
        System.out.println(r+" Row updated successfully ");
    }
    void createTable()throws SQLException{
        String sq = "create table student( USN integer, name varchar(15))";
        int r = stm.executeUpdate(sq);
        System.out.println("Table created successfully.");
    }
    void searchtable()throws SQLException{
        System.out.println("Enter the USN");
        Scanner s = new Scanner(System.in);
        int usn = s.nextInt();
        String a = "SELECT * FROM student where USN="+usn;
        rs = stm.executeQuery(a);
        System.out.println("Student name is "+rs.getString(2));
    }
    void closecon()throws SQLException{
        rs.close();
        stm.close();
        con.close();
    }
}

public class sqlmethods {
    public static void main(String[] args) throws SQLException {
        boolean f = true;
        dbdemo d = new dbdemo();
        Scanner s = new Scanner(System.in);
        int ch;
        do{
            System.out.println("1.Create Table\n2.Display\n3.Insert\n4.Search\n5.Exit\nEnter your Choice");
            ch = s.nextInt();
            switch(ch){
                case 1: d.createTable();
                        break;
                case 2: d.displayinfo();
                        break;
                case 3: d.insertinfo();
                        break;
                case 4: d.searchtable();
                        break;
                case 5: f = false;
                        break;
                default: System.out.println("Invalid choice.");
            }
        }while(f);
        d.closecon();
    }
}
```

17. Create a socket program using TCP sockets to send message between client and server.

TCPServer.java

```
import java.io.*;
import java.net.*;
import java.util.*;
```

```
public class TCPserver{

    private static ServerSocket servSock;

    private static final int PORT = 1234;

    public static void main(String[] args){

        System.out.println("Opening port...\n");

        try{

            servSock = new ServerSocket(PORT);

        }

        catch(IOException ioEx){

            System.out.println("Unable to attach to port!");

            System.exit(1);

        }

        handleClient();

    }

    private static void handleClient(){

        Socket link = null;

        try{

            link = servSock.accept();

            Scanner input = new Scanner(link.getInputStream());

            PrintWriter output = new PrintWriter(link.getOutputStream(),true);

            int numMessages = 0;

            String message = input.nextLine();

            while (!message.equals("close")){

                System.out.println("Message received.");

                numMessages++;
                output.println("Message "  + message);
                message = input.nextLine();
            }
            output.println(numMessages + " messages received.");
        }
        catch(IOException ioEx){
            ioEx.printStackTrace();
        }
        finally{
            try{
                System.out.println("\n* Closing connection... *");
                link.close();
            }
            catch(IOException ioEx){
                System.out.println("Unable to disconnect!");
                System.exit(1);
            }
        }
    }
}
```

TCPClient.java

```
import java.io.*;
import java.net.*;
import java.util.*;
```

```
public class TCPClient{
    private static InetAddress host;
    private static final int PORT = 1234;
    public static void main(String[] args){
        try{
            host = InetAddress.getLocalHost();
        }
        catch(UnknownHostException uhEx){
            System.out.println("Host ID not found!");
            System.exit(1);
        }
        accessServer();
    }
    private static void accessServer(){
        Socket link = null;
        try{
            link = new Socket(host,PORT);
            Scanner input = new Scanner(link.getInputStream());
            PrintWriter output = new PrintWriter( link.getOutputStream(),true);
            Scanner userEntry = new Scanner(System.in);
            String message, response;
            do{
                System.out.print("Enter message: ");
                message = userEntry.nextLine();
                output.println(message);
                response = input.nextLine();

                System.out.println("\nSERVER> "+response);

            }while (!message.equals("close"));

        }

        catch(IOException ioEx){

                                ioEx.printStackTrace();

        }

        finally{

            try{

                System.out.println("\n* Closing connection... *");

                link.close();

            }

            catch(IOException ioEx){

                System.out.println("Unable to disconnect!");

                System.exit(1);

            }

        }

    }

 }
```

18. Create a socket program using UDP sockets to send message between client and server.

UDPServer.java

```
import java.io.*;
import java.net.*;

public class UDPServer{
   private static final int PORT = 1234;
   private static DatagramSocket datagramSocket;
   private static DatagramPacket inPacket, outPacket;
   private static byte[] buffer;
   public static void main(String[] args){
      System.out.println("Opening port...\n");
      try{
         datagramSocket = new DatagramSocket(PORT);
      }
      catch(SocketException sockEx){
         System.out.println("Unable to attach to port!");
         System.exit(1);
      }
      handleClient();
   }
   private static void handleClient(){
      try{
         String messageIn,messageOut;
         int numMessages = 0;
         do{
            buffer = new byte[256];
            inPacket = new DatagramPacket(buffer, buffer.length);
            datagramSocket.receive(inPacket);
            InetAddress clientAddress = inPacket.getAddress();
            int clientPort = inPacket.getPort();
            messageIn = new String(inPacket.getData(), 0,inPacket.getLength());

            System.out.println("Message received.");
            numMessages++;
            messageOut = "Message "  + messageIn;
```

```
        outPacket = new DatagramPacket(messageOut.getBytes(),messageOut.length(),clientAddress, clientPort);

        datagramSocket.send(outPacket);
      }while (true);

    }
    catch(IOException ioEx){
      ioEx.printStackTrace();
    }
    finally{
      System.out.println("\n* Closing connection... *");
      datagramSocket.close();
    }
  }
}
```

UDPClient.java

```
import java.io.*;
import java.net.*;
import java.util.*;

public class UDPClient{
  private static InetAddress host;
  private static final int PORT = 1234;
  private static DatagramSocket datagramSocket;
  private static DatagramPacket inPacket, outPacket;
  private static byte[] buffer;
  public static void main(String[] args){
    try{
      host = InetAddress.getLocalHost();
    }
    catch(UnknownHostException uhEx){
            System.out.println("Host ID not found!");
            System.exit(1);
        }
        accessServer();
    }
    private static void accessServer(){
        try{
            datagramSocket = new DatagramSocket();
            Scanner userEntry = new Scanner(System.in);
            String message="", response="";
            do{
                System.out.print("Enter message: ");
                message = userEntry.nextLine();
                if(!message.equals("close")){
                    outPacket = new DatagramPacket( message.getBytes(), message.length(), host,PORT);
                    datagramSocket.send(outPacket);
                    buffer = new byte[256];
                    inPacket = new DatagramPacket( buffer, buffer.length);
                    datagramSocket.receive(inPacket);
                    response = new String(inPacket.getData(), 0, inPacket.getLength());
                    System.out.println( "\nSERVER> "+response);
                }
            }while (!message.equals("close"));
        }
        catch(IOException ioEx){
            ioEx.printStackTrace();
        }
        finally{
            System.out.println("\n* Closing connection... *");
            datagramSocket.close();
        }
    }
}
```

19. Write a servlet code to show the communication between a web server and client.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

// Extend HttpServlet class
public class HelloWorld extends HttpServlet {

  private String message;

  public void init() throws ServletException
  {
     // Do required initialization
     message = "Hello World";
  }

  public void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException
  {
     // Set response content type
     response.setContentType("text/html");

     // Actual logic goes here.
     PrintWriter out = response.getWriter();
     out.println("<h1>" + message + "</h1>");
  }

  public void destroy()
  {
     // do nothing.
  }
}
```

```
                    <servlet>
   <servlet-name>HelloWorld</servlet-name>
   <servlet-class>HelloWorld</servlet-class>
</servlet>

<servlet-mapping>
   <servlet-name>HelloWorld</servlet-name>
   <url-pattern>/HelloWorld</url-pattern>
</servlet-mapping>
```

20. Write a JSP code to show the communication between a web server and client.

```
<%! int day = 3; %>

<html>
<head><title>SWITCH...CASE Example</title></head>
<body>
<%
switch(day) {
case 0:
  out.println("It\'s Sunday.");
  break;
case 1:
  out.println("It\'s Monday.");
  break;
case 2:
  out.println("It\'s Tuesday.");
  break;
case 3:
  out.println("It\'s Wednesday.");
  break;
case 4:
  out.println("It\'s Thursday.");
  break;
case 5:
  out.println("It\'s Friday.");
  break;
default:
  out.println("It\'s Saturday.");
}
%>
</body>
</html>
```